Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	ST7
Core Size	8-Bit
Speed	8MHz
Connectivity	SPI
Peripherals	PWM, WDT
Number of I/O	13
Program Memory Size	1.5KB (1.5K x 8)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 5x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	16-SOIC (0.154", 3.90mm Width)
Supplier Device Package	16-SO
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/st7fl09y0mae

Table of Contents

1 DESCRIPTION

The ST7L0x is a member of the ST7 microcontroller family suitable for automotive applications. All ST7 devices are based on a common industry-standard 8-bit core, featuring an enhanced instruction set.

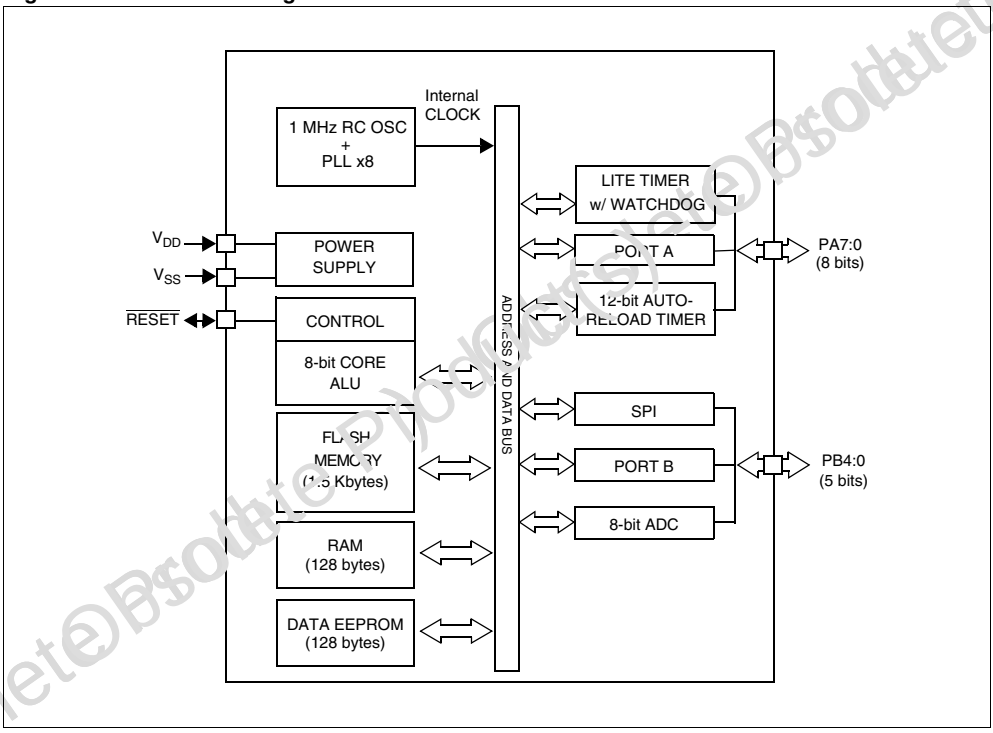
The ST7L0x features Flash memory with byte-by-byte In-Circuit Programming (ICP) and In-Application Programming (IAP) capability.

Under software control, the ST7L0x devices can be placed in WAIT, SLOW, or HALT mode, reducing power consumption when the application is in idle or standby state.

The enhanced instruction set and addressing modes of the ST7 offer both power and flexibility to software developers, enabling the design of highly efficient and compact application code. In addition to standard 8-bit data management, all ST7 microcontrollers feature true bit manipulation, 8 x 8 unsigned multiplication and indirect addressing modes.

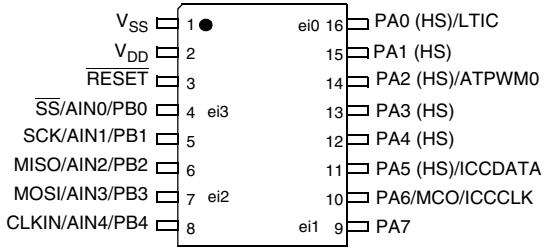
For easy reference, all parametric data is found in Section 13 on page 73.

Figure 1. General Block Diagram



2 PIN DESCRIPTION

Figure 2. 16-Pin Package Pinout (150mil)



(HS) 20mA high sink capability
ei_x associated external interrupt vector

CPU REGISTERS (Cont'd)

CONDITION CODE REGISTER (CC)

Read/Write

Reset Value: 111x1xxx

7							0
1	1	1	H	I	N	Z	C

The 8-bit Condition Code register contains the interrupt mask and four flags representative of the result of the instruction just executed. This register can also be handled by the PUSH and POP instructions.

These bits can be individually tested and/or controlled by specific instructions.

Bit 4 = **H** *Half carry*

This bit is set by hardware when a carry occurs between bits 3 and 4 of the ALU during an ADD or ADC instruction. It is reset by hardware during the same instructions.
0: No half carry has occurred.
1: A half carry has occurred.

This bit is tested using the JRH or JRNH instruction. The H bit is useful in BCD arithmetic subroutines.

Bit 3 = **I** *Interrupt mask*

This bit is set by hardware when entering in interrupt or by software to disable all interrupts except the TRAP software interrupt. This bit is cleared by software.
0: Interrupts are enabled.
1: Interrupts are disabled.

This bit is controlled by the RIM, SIM and IRET instructions and is tested by the JRM and JRNM instructions.

Note: Interrupts requested while I is set are latched and can be processed when I is cleared. By default an interrupt routine is not interruptible because the I bit is set by hardware at the start of the routine and reset by the IRET instruction at the end of the routine. If the I bit is cleared by software in the interrupt routine, pending interrupts are serviced regardless of the priority level of the current interrupt routine.

Bit 2 = **N** *Negative*

This bit is set and cleared by hardware. It is representative of the result sign of the last arithmetic, logical or data manipulation. It is a copy of the 7th bit of the result.
0: The result of the last operation is positive or null.
1: The result of the last operation is negative (that is, the most significant bit is a logic 1).

This bit is accessed by the JRMI and JRPL instructions.

Bit 1 = **Z** *Zero*

This bit is set and cleared by hardware. This bit indicates that the result of the last arithmetic, logical or data manipulation is zero.
0: The result of the last operation is different from zero.
1: The result of the last operation is zero.

This bit is accessed by the JREQ and JRNE test instructions.

Bit 0 = **C** *Carry/borrow*

This bit is set and cleared by hardware and software. It indicates an overflow or an underflow has occurred during the last arithmetic operation.
0: No overflow or underflow has occurred.
1: An overflow or underflow has occurred.

This bit is driven by the SCF and RCF instructions and tested by the JRC and JRNC instructions. It is also affected by the "bit test and branch", shift and rotate instructions.

SUPPLY, RESET AND CLOCK MANAGEMENT (Cont'd)

7.3 REGISTER DESCRIPTION

MAIN CLOCK CONTROL/STATUS REGISTER (MCCSR)
Read / Write
Reset Value: 0000 0000 (00h)

7							0
0	0	0	0	0	0	MCO	SMS

Bits 7:2 = Reserved, must be kept cleared.

Bit 1 = **MCO** *Main Clock Out enable*
This bit is read/write by software and cleared by hardware after a reset. This bit allows to enable the MCO output clock.
0: MCO clock disabled, I/O port free for general purpose I/O.
1: MCO clock enabled.

Bit 0 = **SMS** *Slow Mode select*
This bit is read/write by software and cleared by hardware after a reset. This bit selects the input clock f_{OSC} or $f_{OSC}/32$.
0: Normal mode ($f_{CPU} = f_{OSC}$)
1: Slow mode ($f_{CPU} = f_{OSC}/32$)

RC CONTROL REGISTER (RCCR)
Read / Write
Reset Value: 1111 1111 (FFh)

7							0
CR7	CR6	CR5	CR4	CR3	CR2	CR1	CR0

Bits 7:0 = **CR[7:0]** *RC Oscillator Frequency Adjustment Bits*
These bits must be written immediately after reset to adjust the RC oscillator frequency and to obtain an accuracy of 1%. The application can store the correct value for each voltage range in EEPROM and write it to this register at start-up.
00h = maximum available frequency
FFh = lowest available frequency

Note: To tune the oscillator, write a series of different values in the register until the correct frequency is reached. The fastest method is to use a dichotomy starting with 80h.

SYSTEM INTEGRITY (SI) CONTROL/STATUS REGISTER (SICSR)
Read/Write
Reset Value: 0000 0x00 (0xh)

7							0
0	0	0	0	LOCKED	0	0	0

Bits 7:4 = Reserved, must be kept cleared.

Bit 3 = **LOCKED** *PLL Locked Flag*
This bit is set and cleared by hardware. It is set automatically when the PLL reaches its operating frequency.
0: PLL not locked
1: PLL locked

Bits 2:0 = Reserved, must be kept cleared.

Table 3. Clock Register Map and Reset Values

Address (Hex.)	Register Label	7	6	5	4	3	2	1	0
0038h	**MCCSR** Reset Value	0	0	0	0	0	0	MCO 0	SMS 0
0039h	**RCCR** Reset Value	CR7 1	CR6 1	CR5 1	CR4 1	CR3 1	CR2 1	CR1 1	CR0 1
003Ah	**SICSR** Reset Value	0	0	0	0	LOCKED 0	0	0	0

SUPPLY, RESET AND CLOCK MANAGEMENT (Cont'd)

7.4 RESET SEQUENCE MANAGER (RSM)

7.4.1 Introduction

The reset sequence manager includes two RESET sources as shown in Figure 14:

- External $\overline{\text{RESET}}$ source pulse
- Internal WATCHDOG RESET

Note: A reset can also be triggered following the detection of an illegal opcode or prebyte code. Refer to Section 12.2.1 on page 70 for further details.

These sources act on the $\overline{\text{RESET}}$ pin and it is always kept low during the delay phase.

The RESET service routine vector is fixed at addresses FFFEh-FFFFh in the ST7 memory map.

The basic RESET sequence consists of three phases as shown in Figure 13:

- Active Phase depending on the RESET source
- 256 CPU clock cycle delay
- RESET vector fetch

Caution: When the ST7 is unprogrammed or fully erased, the Flash is blank and the RESET vector is not programmed. For this reason, it is recommended to keep the $\overline{\text{RESET}}$ pin in low state until programming mode is entered, in order to avoid unwanted behavior.

The 256 CPU clock cycle delay allows the oscillator to stabilize and ensures that recovery has taken place from the Reset state.

The RESET vector fetch phase duration is two clock cycles.

If the PLL is enabled by option byte, it outputs the clock after an additional delay of $t_{STARTUP}$ (see Figure 11).

Figure 13. RESET Sequence Phases

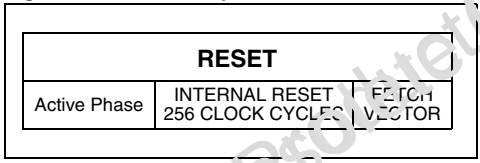
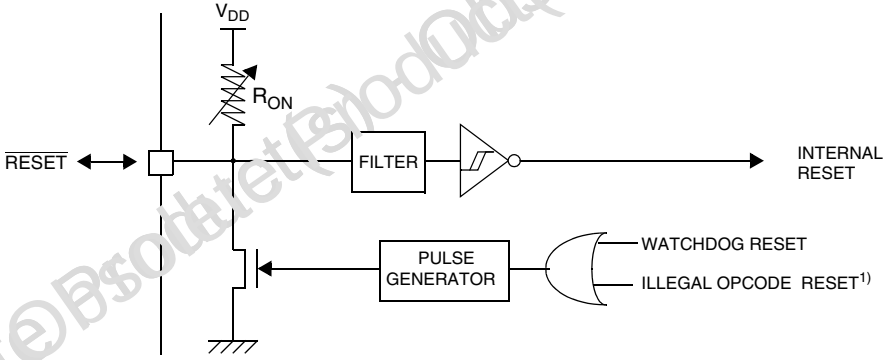


Figure 14. ST7L0x Reset Block Diagram



Note 1: See "Illegal Opcode Reset" on page 70. for more details on illegal opcode reset conditions.

POWER SAVING MODES (Cont'd)

9.4.2.1 HALT Mode Recommendations

– Make sure that an external event is available to wake up the microcontroller from Halt mode.

– When using an external interrupt to wake up the microcontroller, reinitialize the corresponding I/O as "Input Pull-up with Interrupt" before executing the HALT instruction. The main reason for this is that the I/O may be wrongly configured due to external interference or by an unforeseen logical condition.

– For the same reason, reinitialize the level sensitiveness of each external interrupt as a precautionary measure.

– The opcode for the HALT instruction is 0x8E. To avoid an unexpected HALT instruction due to a program counter failure, it is advised to clear all occurrences of the data value 0x8E from memory. For example, avoid defining a constant in ROM with the value 0x8E.

– As the HALT instruction clears the I bit in the CC register to allow interrupts, the user may choose to clear all pending interrupt bits before executing the HALT instruction. This avoids entering other peripheral interrupt routines after executing the external interrupt routine corresponding to the wake-up event (reset or external interrupt).

register

– select rising edge

reset the interrupt mask with the RIM instruction (in cases where a pin level change could occur)

10.2.2 Output Modes

The output configuration is selected by setting the corresponding DDR register bit. In this case, writing the DR register applies this digital value to the I/O pin through the latch. Then reading the DR register returns the previously stored value.

Two different output modes can be selected by software through the OR register: Output push-pull and open-drain.

DR register value and output pin status:

DR	Push-pull	Open-drain
0	V_{SS}	Vss
1	V_{DD}	Floating

Note: When switching from input to output mode, the DR register must be written first to drive the correct level on the pin as soon as the port is configured as an output.

10.2.3 Alternate Functions

When an on-chip peripheral is configured to use a pin, the alternate function is automatically selected. This alternate function takes priority over the standard I/O programming under the following conditions:

– When the signal is coming from an on-chip peripheral, the I/O pin is automatically configured in output mode (push-pull or open drain according to the peripheral).

– When the signal is going to an on-chip peripheral, the I/O pin must be configured in floating input mode. In this case, the pin state is also digitally readable by addressing the DR register.

Notes:

– Input pull-up configuration can cause unexpected value at the input of the alternate peripheral input.

– When an on-chip peripheral uses a pin as input and output, this pin must be configured in input floating mode.

12-BIT AUTORELOAD TIMER (Cont'd)

AUTORELOAD REGISTER (ATRH)
Read / Write
Reset Value: 0000 0000 (00h)

15							8
0	0	0	0	ATR11	ATR10	ATR9	ATR8

AUTORELOAD REGISTER (ATRL)
Read / Write
Reset Value: 0000 0000 (00h)

7							0
ATR7	ATR6	ATR5	ATR4	ATR3	ATR2	ATR1	ATR0

Bits 15:12 = Reserved, must be kept cleared.

Bits 11:0 = **ATR[11:0]** *Autoreload Register*
This is a 12-bit register which is written by software. The ATR register value is automatically loaded into the upcounter when an overflow occurs. The register value is used to set the PWM frequency.

PWM0 DUTY CYCLE REGISTER HIGH (DCR0H)
Read / Write
Reset Value: 0000 0000 (00h)

15							8
0	0	0	0	DCR11	DCR10	DCR9	DCR8

PWM0 DUTY CYCLE REGISTER LOW (DCR0L)
Read / Write
Reset Value: 0000 0000 (00h)

7							0
DCR7	DCR6	DCR5	DCR4	DCR3	DCR2	DCR1	DCR0

Bits 15:12 = Reserved, must be kept cleared.

Bits 11:0 = **DCR[11:0]** *PWMx Duty Cycle Value*
This 12-bit value is written by software. The high register must be written first.

In PWM mode (OE0 = 1 in the PWMCR register) the DCR[11:0] bits define the duty cycle of the PWM0 output signal (see Figure 29). In Output Compare mode, (OE0 = 0 in the PWMCR register) they define the value to be compared with the 12-bit upcounter value.

PWM0 CONTROL/STATUS REGISTER (PWM0CSR)
Read / Write
Reset Value: 0000 0000 (00h)

7							0
0	0	0	0	0	0	OP0	CMPF0

Bit 7:2 = Reserved, must be kept cleared.

Bit 1 = **OP0** *PWM0 Output Polarity*
This bit is read/write by software and cleared by hardware after a reset. This bit selects the polarity of the PWM0 signal.
0: The PWM0 signal is not inverted.
1: The PWM0 signal is inverted.

Bit 0 = **CMPF0** PWM0 *Compare Flag.*
This bit is set by hardware and cleared by software by reading the PWM0CSR register. It indicates that the upcounter value matches the DCR0 register value.
0: Upcounter value does not match DCR value.
1: Upcounter value matches DCR value.

SERIAL PERIPHERAL INTERFACE (Cont'd)

11.3.3.1 Functional Description

A basic example of interconnections between a single master and a single slave is illustrated in Figure 32.

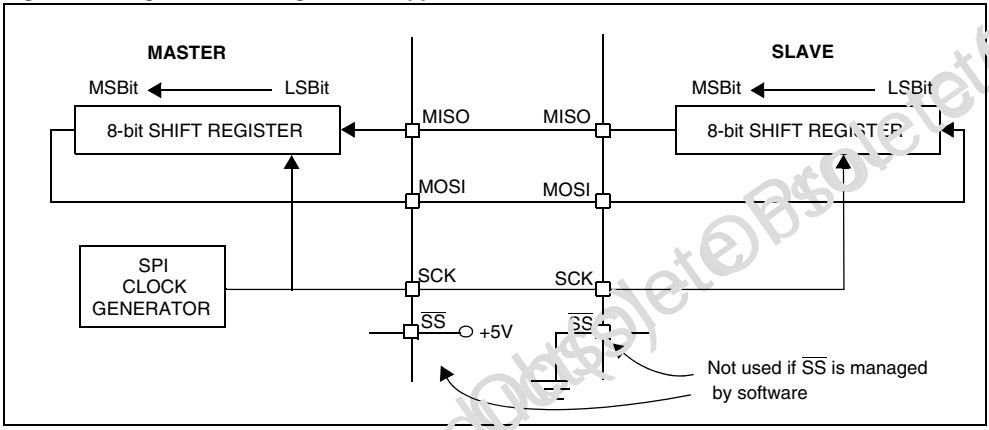
The MOSI pins are connected together and the MISO pins are connected together. In this way data is transferred serially between master and slave (most significant bit first).

The communication is always initiated by the master. When the master device transmits data to a slave device via MOSI pin, the slave device responds by sending data to the master device via the MISO pin. This implies full duplex communication with both data out and data in synchronized with the same clock signal (which is provided by the master device via the SCK pin).

To use a single data line, the MISO and MOSI pins must be connected at each node (in this case only simplex communication is possible).

Four possible data/clock timing relationships may be chosen (see Figure 35) but master and slave must be programmed with the same timing mode.

Figure 32. Single Master/ Single Slave Application



SERIAL PERIPHERAL INTERFACE (Cont'd)

11.3.6 Low Power Modes

Mode	Description
WAIT	No effect on SPI. SPI interrupt events cause the device to exit from WAIT mode.
HALT	SPI registers are frozen. In HALT mode, the SPI is inactive. SPI operation resumes when the MCU is woken up by an interrupt with "exit from HALT mode" capability. The data received is subsequently read from the SPIDR register when the software is running (interrupt vector fetching). If data is received before the wake-up event, then an overrun error is generated. This error can be detected after the fetch of the interrupt routine that woke up the device.

11.3.6.1 Using the SPI to wakeup the MCU from Halt mode

In slave configuration, the SPI is able to wakeup the ST7 device from HALT mode through a SPIF interrupt. The data received is subsequently read from the SPIDR register when the software is running (interrupt vector fetch). If multiple data transfers have been performed before software clears the SPIF bit, then the OVR bit is set by hardware.

Note: When waking up from Halt mode, if the SPI remains in Slave mode, it is recommended to perform an extra communications cycle to bring the SPI from Halt mode state to normal state. If the SPI exits from Slave mode, it returns to normal state immediately.

Caution: The SPI can wake up the ST7 from Halt mode only if the Slave Select signal (external $\overline{SS}$ pin or the SSI bit in the SPICSR register) is low when the ST7 enters Halt mode. So if Slave selection is configured as external (see Section 11.3.3.2), make sure the master drives a low level on the $\overline{SS}$ pin when the slave enters Halt mode.

11.3.7 Interrupts

Interrupt Event	Event Flag	Enable Control Bit	Exit from Wait	Exit from Halt
SPI End of Transfer Event	SPIF	SPIE	Yes	Yes
Master Mode Fault Event	MODF			No
Overrun Error	OVR			

Note: The SPI interrupt events are connected to the same interrupt vector (see Interrupts chapter). They generate an interrupt if the corresponding Enable Control Bit is set and the interrupt mask in the CC register is reset (RIM instruction).

ST7 ADDRESSING MODES (Cont'd)

12.1.6 Indirect Indexed (Short, Long)

This is a combination of indirect and short indexed addressing modes. The operand is referenced by its memory address, which is defined by the unsigned addition of an index register value (X or Y) with a pointer value located in memory. The pointer address follows the opcode.

The indirect indexed addressing mode consists of two submodes:

Indirect Indexed (Short)

The pointer address is a byte, the pointer size is a byte, thus allowing 00 - 1FE addressing space, and requires 1 byte after the opcode.

Indirect Indexed (Long)

The pointer address is a byte, the pointer size is a word, thus allowing 64 Kbyte addressing space, and requires 1 byte after the opcode.

Table 19. Instructions Supporting Direct, Indexed, Indirect and Indirect Indexed Addressing Modes

Long and Short Instructions	Function
LD	Load
CP	Compare
AND, OR, XOR	Logical Operations
ADC, ADD, SUB, SBC	Arithmetic Addition/subtraction operations
BCP	Bit Compare

Short Instructions Only	Function
CLR	Clear
INC, DEC	Increment/Decrement
TNZ	Test Negative or Zero
CPL, NEG	1 or 2 Complement
BSET, BRES	Bit Operations
BTJT, BTJF	Bit Test and Jump Operations
SLL, SRL, SRA, RLC, RRC	Shift and Rotate Operations
SWAP	Swap Nibbles
CALL, JP	Call or Jump subroutine

12.1.7 Relative Mode (Direct, Indirect)

This addressing mode is used to modify the PC register value by adding an 8-bit signed offset to it.

Available Relative Direct/ Indirect Instructions	Function
JRxx	Conditional Jump
CALLR	Call Relative

The relative addressing mode consists of two submodes:

Relative (Direct)

The offset follows the opcode.

Relative (Indirect)

The offset is defined in memory, of which the address follows the opcode.

INSTRUCTION GROUPS (Cont'd)

Mnemo	Description	Function/Example	Dst	Src	H	I	N	Z	C
JRULE	Jump if (C + Z = 1)	Unsigned <=							
LD	Load	dst <= src	reg, M	M, reg			N	Z	
MUL	Multiply	X,A = X * A	A, X, Y	X, Y, A	0				0
NEG	Negate (2's compl)	neg $10	reg, M				N	Z	C
NOP	No Operation								
OR	OR operation	A = A + M	A	M			N	Z	
POP	Pop from the Stack	pop reg	reg	M					
		pop CC	CC	M	H	I	N	Z	C
PUSH	Push onto the Stack	push Y	M	reg, CC					
RCF	Reset carry flag	C = 0							0
RET	Subroutine Return								
RIM	Enable Interrupts	I = 0				0			
RLC	Rotate left true C	C <= Dst <= C	reg, M				N	Z	C
RRC	Rotate right true C	C => Dst => C	reg, M				N	Z	C
RSP	Reset Stack Pointer	S = Max allowed							
SBC	Subtract with Carry	A = A - M - C	A	M			N	Z	C
SCF	Set carry flag	C = 1							1
SIM	Disable Interrupts	I = 1				1			
SLA	Shift left Arithmetic	C <= Dst <= 0	reg, M				N	Z	C
SLL	Shift left Logic	C <= Dst <= 0	reg, M				N	Z	C
SRL	Shift right Logic	0 => Dst => C	reg, M				0	Z	C
SRA	Shift right Arithmetic	Dst[7] => Dst => C	reg, M				N	Z	C
SUB	Subtraction	A = A - M	A	M			N	Z	C
SWAP	SWAP nibbles	Dst[7..4] <=> Dst[3..0]	reg, M				N	Z	
TNZ	Test for Neg & Zero	tnz lbl1					N	Z	
TRAP	S/W trap	S/W interrupt				1			
WFI	Wait for Interrupt					0			
XOR	Exclusive OR	A = A XOR M	A	M			N	Z	

13.7 EMC CHARACTERISTICS

Susceptibility tests are performed on a sample basis during product characterization.

13.7.1 Functional EMS (Electro Magnetic Susceptibility)

Based on a simple running application on the product (toggling 2 LEDs through I/O ports), the product is stressed by two electro magnetic events until a failure occurs (indicated by the LEDs).

- **ESD**: Electro-Static Discharge (positive and negative) is applied on all pins of the device until a functional disturbance occurs. This test conforms with the IEC 1000-4-2 standard.
- **FTB**: A Burst of Fast Transient voltage (positive and negative) is applied to V_{DD} and V_{SS} through a 100pF capacitor, until a functional disturbance occurs. This test conforms with the IEC 1000-4-4 standard.

A device reset allows normal operations to resume. The test results are given in the table below based on the EMS levels and classes defined in application note AN1709.

13.7.1.1 Designing Hardened Software to Avoid Noise Problems

EMC characterization and optimization are performed at component level with a typical application environment and simplified MCU software. It should be noted that good EMC performance is highly dependent on the user application and the software in particular.

Therefore it is recommended that the user applies EMC software optimization and prequalification tests in relation with the EMC level requested for his application.

Software recommendations:

The software flowchart must include the management of runaway conditions such as:

– Corrupted program counter

– Unexpected reset

– Critical Data corruption (control registers...)

Prequalification trials:

Most of the common failures (unexpected reset and program counter corruption) can be reproduced by manually forcing a low state on the RESET pin or the Oscillator pins for 1 second.

To complete these trials, ESD stress can be applied directly on the device, over the range of specification values. When unexpected behavior is detected, the software can be hardened to prevent unrecoverable errors occurring (see application note AN1015).

Symbol	Parameter	Conditions	Level/ Class
V_{FESD}	Voltage limits to be applied on any I/O pin to induce a functional disturbance	V_{DD} = 5V, T_A = 25°C, f_{OSC} = 8 MHz conforms to IEC 1000-4-2	2B
V_{FFTB}	Fast transient voltage burst limits to be applied through 100pF on V_{DD} and V_{DD} pins to induce a functional disturbance	V_{DD} = 5V, T_A = 25°C, f_{OSC} = 8 MHz conforms to IEC 1000-4-4	3B

13.7.2 Electro Magnetic Interference (EMI)

Based on a simple application running on the product (toggling two LEDs through the I/O ports), the product is monitored in terms of emission. This emission test is in line with the norm SAE J 1752/3 which specifies the board and the loading of each pin.

Symbol	Parameter	Conditions	Monitored Frequency Band	Max vs [f_{OSC}/f_{CPU}]		Unit
				1/4 MHz	1/8 MHz	
S_{EMI}	Peak level[1)]	V_{DD} = 5V, T_A = 25°C, SO16 package, conforming to SAE J 1752/3	0.1 MHz to 30 MHz	8	14	dBµV
			30 MHz to 130 MHz	27	32	
			130 MHz to 1 GHz	26	28	
			SAE EMI Level	3.5	4	-

Notes:

1. Data based on characterization results, not tested in production.

COMMUNICATION INTERFACE CHARACTERISTICS (Cont'd)

Figure 66. SPI Slave Timing Diagram with CPHA = 1[1)]

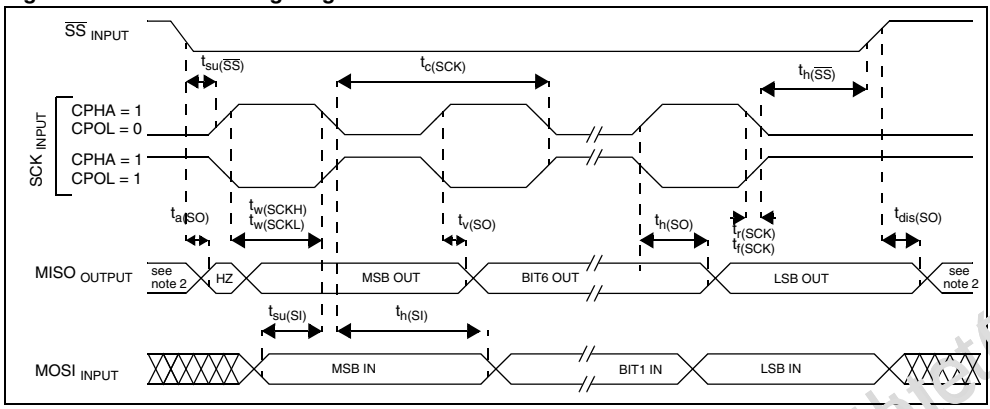
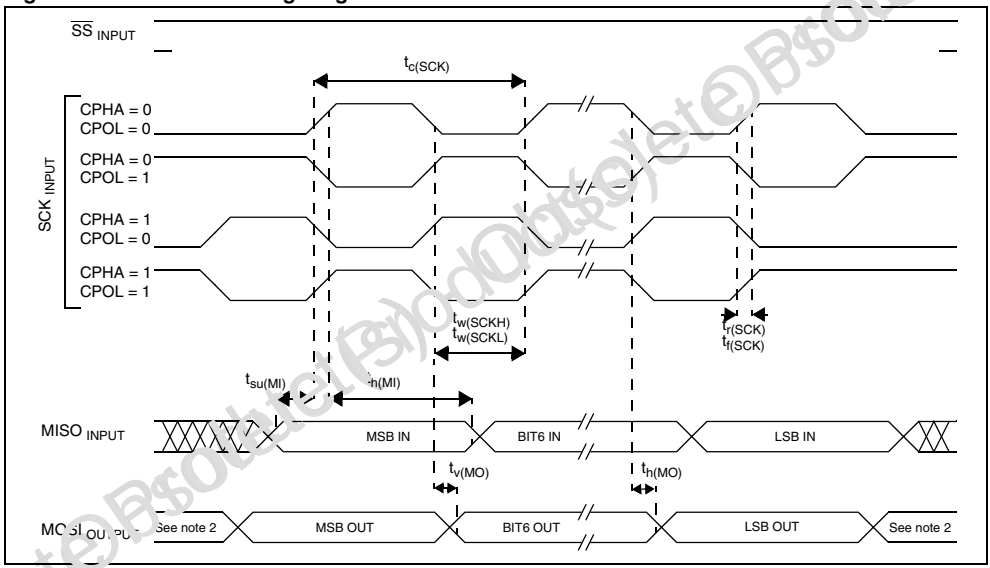


Figure 67. SPI Master Timing Diagram[1)]



Notes:

1. Measurement points are done at CMOS levels: $0.3xV_{DD}$ and $0.7xV_{DD}$.

2. When no communication is on-going the data output line of the SPI (MOSI in master mode, MISO in slave mode) has its alternate function capability released. In this case, the pin status depends of the I/O port configuration.

ADC CHARACTERISTICS (Cont'd)

Figure 69. R_{AIN} max. vs f_{ADC} with C_{AIN} = 0pF[1)]

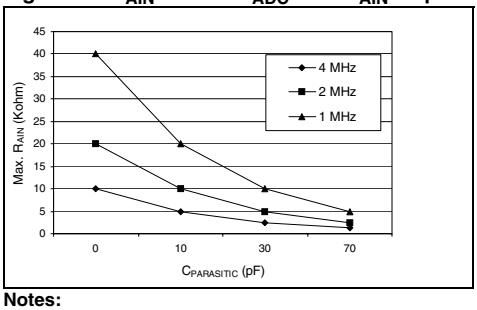
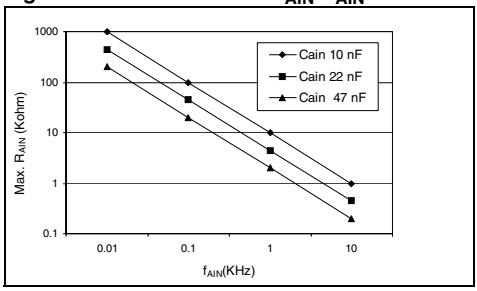


Figure 70. Recommended C_{AIN}/R_{AIN} values[2)]



Notes:

1.$C_{PARASITIC}$ represents the capacitance of the PCB (dependent on soldering and PCB layout quality) plus the pad capacitance (3pF). A high $C_{PARASITIC}$ value will downgrade conversion accuracy. To remedy this, f_{ADC} must be reduced.

2. This graph shows that depending on the input signal variation (f_{AIN}), C_{AIN} can be increased for stabilization and to allow the use of a larger serial resistor (R_{AIN}). It is valid for all f_{ADC} frequencies ≤ 4 MHz.

13.11.0.1 General PCB Design Guidelines

To obtain best results, some general design and layout rules must be followed when designing the application PCB to shield the noise-sensitive, analog physical interface from noise-generating CMOS logic signals.

– Properly place components and route the signal traces on the PCB to shield the analog inputs. Analog signals paths should run over the analog ground plane and be as short as possible. Isolate analog signals from digital signals that may switch while the analog inputs are being sampled by the A/D converter. Do not toggle digital outputs on the same I/O port as the A/D input being converted.

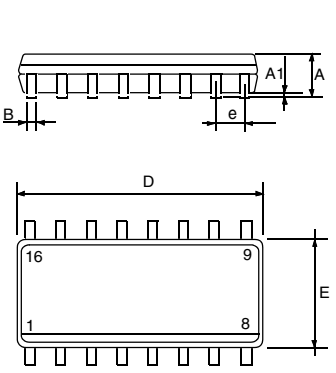
14 PACKAGE CHARACTERISTICS

In order to meet environmental requirements, ST offers these devices in ECOPACK® packages. These packages have a lead-free second level interconnect. The category of second Level Interconnect is marked on the package and on the inner box label, in compliance with JEDEC Standard JESD97. The maximum ratings related to soldering conditions are also marked on the inner box label.

ECOPACK is an ST trademark. ECOPACK specifications are available at www.st.com.

14.1 PACKAGE MECHANICAL DATA

Figure 72. 16-Pin Plastic Small Outline Package, 150-mil Width



0016020

Dim.	mm			inches		
	Min	Typ	Max	Min	Typ	Max
A	1.35		1.75	0.053		0.069
A1	0.10		0.25	0.004		0.010
B	0.33		0.51	0.013		0.020
C	0.19		0.25	0.007		0.010
D	9.80		10.00	0.386		0.394
E	3.80		4.00	0.150		0.157
e		1.27			0.050	
H	5.80		6.20	0.228		0.244
α	0°		8°	0°		8°
L	0.40		1.27	0.016		0.050
	Number of Pins					
N	16					

14.2 THERMAL CHARACTERISTICS

Symbol	Ratings	Value	Unit
R_{thJA}	Package thermal resistance (junction to ambient)	95	°C/W
T_{Jmax}	Maximum junction temperature[1]	150	°C
P_{Dmax}	Power dissipation[2]	500	mW

Notes:
1. The maximum chip-junction temperature is based on technology characteristics.
2. The maximum power dissipation is obtained from the formula $P_D = (T_J - T_A) / R_{thJA}$.
The power dissipation of an application can be defined by the user with the formula: $P_D = P_{INT} + P_{PORT}$ where P_{INT} is the chip internal power (I_{DD} x V_{DD}) and P_{PORT} is the port power dissipation depending on the ports used in the application.

PACKAGE CHARACTERISTICS (Cont'd)

14.3 SOLDERING INFORMATION

In accordance with the RoHS European directive, all STMicroelectronics packages have been converted to lead-free technology named ECOPACK™.

- ECOPACK™ packages are qualified according to the JEDEC STD-020C compliant soldering profile.
- Detailed information on the STMicroelectronics ECOPACK™ transition program is available on www.st.com/stonline/leadfree/, with specific technical application notes covering the main technical aspects related to lead-free conversion (AN2033, AN2034, AN2035, and AN2036).

Forward compatibility

ECOPACK™ SO packages are fully compatible with a lead (Pb) containing soldering process (see application note AN2034).

Table 22. Soldering Compatibility (Wave and Reflow Soldering Process)

Package	Plating material	Pb solder paste	Pb-free solder paste
SO	NiPdAu (Nickel-Palladium-Gold)	Yes	Yes

DEVICE CONFIGURATION AND ORDERING INFORMATION (Cont'd)

OPTION BYTE 1

Bit 7 = Reserved, must always be 1

Bit 6 = **PLLOFF** *PLL disabled*
0: PLL enabled
1: PLL disabled (by-passed)

Bit 5 = Reserved, must always be 1

Bit 4 = **OSC** *RC Oscillator selection*
0: RC oscillator on
1: RC oscillator off

Note: If the RC oscillator is selected, then to improve clock stability and frequency accuracy, it is recommended to place a decoupling capacitor, typically 100nF, between the V_{DD} and V_{SS} pins as close as possible to the ST7 device.

Table 23. List of Valid Option Combinations

Operating conditions				Option Bits	
V_{DD} range	Clock Source	PLL	Typ f_{CPU}	OSC	PLLOFF
3.6 to 5.5V	Internal RC 1%	off	1 MHz @ 5V	0	1
		x8	8 MHz @ 5V		0
	External clock	off	0 to 8 MHz	1	1
		x8	8 MHz		0

Note: See Clock Management Block diagram in Figure 12**.**

Bits 3:2 = Reserved

Bit 1 = **WDG SW** *Hardware or software watchdog*
This option bit selects the watchdog type.
0: Hardware (watchdog always enabled)
1: Software (watchdog to be enabled by software)

Bit 0 = **WDG HALT** *Watchdog Reset on Halt*
This option bit determines if a RESET is generated when entering HALT mode while the Watchdog is active.
0: No Reset generation when entering Halt mode
1: Reset generation when entering Halt mode

	OPTION BYTE 0								OPTION BYTE 1							
	7							0	7							0
	Reserved				SEC1	SEC0	FMP R	FMP W	Res.	PLL OFF	Res.	OSC	Res.	Res.	WDG SW	WDG HALT
Default Value	1	1	1	1	1	1	0	0	1	1	1	0	1	1	1	1

DEVICE CONFIGURATION AND ORDERING INFORMATION (Cont'd)

Figure 74. FASTROM Commercial Product Code Structure

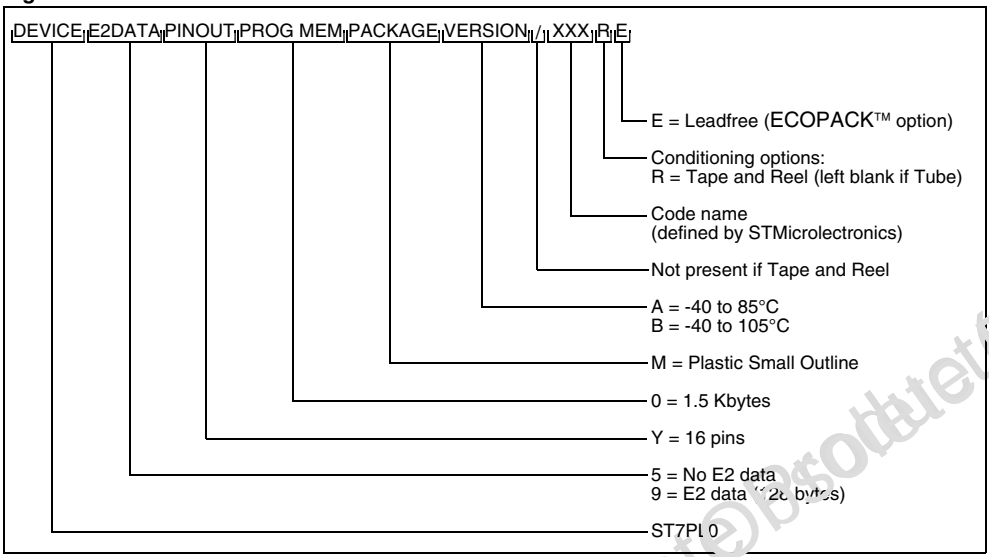


Table 25. FASTROM Factory Coded Device Types

Part Number	Program Memory (Bytes)	Data EEPROM (Bytes)	RAM (Bytes)	Temp. Range	Package
ST7PL05Y0MA	1.5K FASTROM	-	128	-40 to +85°C	SO16
ST7PL09Y0MA		128			
ST7PL05Y0MB		-		-40 to +105°C	
ST7PL09Y0MB		128			